

Scène initiale: Lucien va à l'événement dont Sarah, la fille qu'il a rencontré au bar, lui a parlé.

1 - Workshop sur les softwares

Scénario : L'évènement est divisé en plusieurs parties. La première est un workshop sur les softwares utilisés par les producteurs. Une personne va parler dans une salle et présenter les softwares.

Déclencheur : Entrer dans la salle des workshops

Exercice : À la fin du workshop, une fille qui est arrivée en retard, lui demande de résumer les outils qui ont été mentionnés. Lier les outils aux fonctionnalités.

Ecrans : 1 avec la salle, 1 avec la personne qui parle, 1 avec la fille qui pose la question à Lucien.

Musique : Peut-être des sons créés avec les softwares.

2 - Conférence sur les nouveaux défis du marché

Scénario : La deuxième partie est une conférence. Une personne parle des nouveaux défis du marché de la production de musique pour les jeux.

Déclencheur : Entre dans la salle de la conférence.

Exercice : La personne qui parle va présenter les défis, mais ils ne les dit pas dans la bonne ordre. (il y a un problème avec les slides, peut-être). Il faudra, à la fin, les mettre dans le bon ordre d'importance (selon ce qui dit le présentateur).

Ecrans :

On explique les défis du marché aujourd'hui. Trier les défis en ordre d'importance (l'ordre que la personne qui parle va donner)

1. Adaptive & Interactive Music Systems

The challenge:

Music isn't just "background" anymore — it must respond dynamically to gameplay (e.g., transitions, layers, player choices).

Why it's hard:

- You must compose *not just tracks*, but modules, stems, cues that can shift smoothly.
- Implementation requires fluency with middleware like FMOD or Wwise and close work with programmers.

Result:

Producers spend more time on structure and logic than on linear composition.

2. Cross-disciplinary Collaboration

The challenge:

Game music producers must communicate with designers, programmers, audio directors, and QA.

Why it's hard:

- Different teams use different jargon and pipelines.
- Music often must align with art, narrative, and gameplay timing.

Result:

Strong soft skills — documentation, flexibility, and teamwork — are as important as musical ability.

3. Ever-higher Quality Expectations

The challenge:

With big budgets and powerful hardware, players expect cinematic scores, lush soundscapes, and deep sound design even in indie games.

Why it's hard:

- More time (and skill) needed for orchestration, live recording, mixing, and mastering.
- Smaller teams may struggle to match AAA quality.

Result:

Music producers must wear multiple hats (composer, mixer, engineer, sometimes performer).



4. Balancing Creativity & Memory/Performance Constraints

The challenge:

Even in 2026, games must run on many platforms (PC, consoles, handheld/streaming), each with memory and performance limits.

Why it's hard:

- Audio assets (samples, stems) can be large.
- Adaptive systems multiply the number of cues.

Result:

Producers must optimize music and work closely with audio programmers on compression, streaming, and architecture.



5. Global & Cultural Sensitivity

The challenge:

Games reach players worldwide — soundtracks must resonate culturally without stereotyping or alienation.

Why it's hard:

- Incorporating authentic styles requires research, collaboration, or consultation.
- Missteps can feel tone-deaf or disrespectful.

Result:

Producers increasingly work with specialists, ethnomusicologists, or cultural consultants.



6. AI Tools — Opportunity & Ethical Dilemma

The challenge:

AI composition and sound design tools are powerful (quick sketches, variant generation), but raise questions:

- Originality: Are AI-generated ideas truly creative?
- Copyright: Who owns AI-assisted music?
- Skill relevance: Do producers risk becoming “button-pushers”?

Why it's hard:

Integrating AI effectively *without* losing artistic voice or facing legal pitfalls.

Result:

Producers must stay technically literate and ethically informed.

7. Monetization & Visibility

The challenge:

Standing out as a game music producer — especially indie — can be hard.

Why it's hard:

- Streaming algorithms don't always favor interactive or adaptive music formats.
- Traditional platforms (YouTube, Spotify) don't reflect *in-game experience*.

Result:

Producers build personal brands, portfolios, and community presence.

8. Maintenance & Post-Launch Support

The challenge:

Games are now services: updates, expansions, seasonal events.

Why it's hard:

Music may need new layers, themes, or adaptive logic post-launch, often on tight timelines.

Result:

Producers support a game long after release, not just at launch.

3 - Talk d'un producteur qui utilise l'IA

Scénario :

Déclencheur :

Exercice :

Ecrans :

Metro Loopin dit qu'il fait une grosse partie de son travail avec l'IA, parce qu'il s'en fout et de toute façon, pour un jeu vidéo, personne va s'en apercevoir. Il explique comment utiliser l'IA:

1. AI for ideation & sketching (early creative phase)

What AI does

- Generates musical sketches, chord progressions, motifs
- Creates multiple stylistic variations quickly
- Helps break creative blocks

How producers use it

- Generate 10–20 rough ideas in minutes
- Pick 1–2 and rewrite them manually in a DAW
- Use AI output as *raw material*, not final music

Typical tools

- AIVA
- Soundraw

✓ Good for: prototyping, temp tracks

✗ Risk: generic or emotionally flat music if used blindly

2. AI for adaptive music variations

What AI does

- Generates micro-variations (rhythm, harmony, density)
- Helps avoid repetition in looping game music

How producers use it

- Feed a main theme → get subtle alternates
- Use variations inside FMOD/Wwise as layers or states
- Maintain cohesion while increasing playtime longevity

This is powerful for:

- open-world games
- roguelikes
- live-service titles

3. AI for sound design & audio assets

What AI does


- Generates textures, ambiences, abstract sounds
- Assists in creating large sound libraries quickly

How producers use it

- Create raw material → edit, process, and curate
- Combine AI sounds with traditional synthesis and foley

Typical use cases

- sci-fi environments
- surreal / experimental worlds
- placeholder assets during development

 Always clean, edit, and contextualize — raw AI audio rarely works in-game.

4. AI for technical & workflow tasks

What AI does

- Automates repetitive work
- Assists with implementation logic and scripting

Examples

- Generate FMOD/Wwise logic ideas
- Suggest parameter mappings (intensity, tension, speed)
- Help write documentation for dev teams
- Convert design notes into implementation checklists

This saves hours, not creativity.

5. AI for mixing, mastering & optimization

What AI does

- Loudness balancing
- EQ suggestions
- Platform-specific optimization (console vs mobile)

How producers use it

- As a second opinion, not final authority
- Especially useful for fast iteration and QA builds

⚠ Game audio still needs human judgment because:

- music is interactive
- loudness changes with gameplay
- context matters more than polish

6. AI for testing & player response simulation

Emerging but growing use:

- Simulate long play sessions
- Test whether music becomes annoying or repetitive
- Analyze transitions and pacing problems

This helps producers think in hours of gameplay, not minutes of music.

7. Ethical & professional boundaries (very important)

Smart producers in 2026:

- ✓ Use AI assistively
- ✓ Keep authorship and originality clear
- ✓ Check licensing and training data policies
- ✗ Don't ship fully AI-generated music without control
- ✗ Don't rely on AI for emotional storytelling

Studios increasingly ask:

"Where does the human creativity start?"

You should always have a clear answer.

Choix = justifier l'utilisation de l'IA ou s'énervé ?

Dans le premier cas, tout le monde se montrera contraire à l'utilisation de l'IA, dans le deuxième, tout le monde le supportera.

4 - Networking avec d'autres producteurs

Scénario :

Déclencheur :

Exercice :

Ecrans :

Connaître des nouveaux producteurs. Chacun a produit un jeu décalé: comprendre quels jeux ont-ils créés.

Le jeu du marathon : on choisit la playlist à écouter en courant. La musique accélère ou ralenti en s'adaptant à la vitesse du personnage qui court

Le jeu du ménage : compléter les tâches ménagères. On commence avec notre propre maison, après on travail chez des amis, dans des bibliothèques (attention à ne pas faire du bruit), dans des salles des sports, etc.. - Le producteur a produit les bruits de chaque tâche (machine à laver, balai, sprays, la musique de fond change lorsqu'on monte de niveau)

Loading screen - the game : le jeu ne commence jamais, le joueur s'entretien avec des minijeu lors du chargement de la page. Chaque petit jeu est accompagné par une musique différente

THE CEO : le jeu consiste en gérer une entreprise qui ne produit rien. Il faudra créer des pubs pour ne rien vendre (la musique est ici), employer des travailleurs avec des rôles criptyques "Chief motivation officer", "Proactivity head consultant" et virer les employés avec des phrases motivationnelles.

5 - Conversation avec Sarah

Scénario :

Déclencheur :

Exercice :

Ecrans :

Il parle de sa musique. il défend ses idées avec Sarah qui essaye de le mener dans la mauvaise route. Ex: si Lucien avait justifié l'utilisation de l'IA, Sarah va soutenir que l'IA est terrible; si Lucien n'aime pas l'IA, Sarah va dire que l'IA est un outil fondamental pour produire de la musique aujourd'hui. En tout cas, il doit motiver ses réponses.

Au final on découvre que Sarah n'a pas une idée fixe sur l'IA, mais elle voulait juste le tester pour voir s'il croit à ses convictions.

6 - Conversation avec un membre de l'entreprise

Scénario :

Déclencheur :

Exercice :

Ecrans :

Conversation avec un membre de l'entreprise. Lucien lui fait écouter l'un de ses projets et lui explique comment il travaille (avec quels outils, dans quels moments de la journée, pour combien de temps) et arrive à prendre un rendez-vous avec l'un des responsables des ressources humaines. Il sera en ligne dans quelques jours.

Scène finale

Déclencheur :

Ecrans :